8.3 Extensibility Rules

In accordance with ISO 8824-1 Amendment 1 on extensibility rules, the productions that are of extensible types are to be indicated by including the following in their type descriptions ...

The following types will be indicated as being extensible:

- ENUMERATED
- named INTEGER
- named BIT STRING
- SET
- SEQUENCE
- CHOICE

Under the rules of extensibility new enumerations (for ENUMERATED types), new bit name assignments (for named BIT STRING types), new named numbers (for named INTEGER types), and new elements (for SET, SEQUENCE, and CHOICE types) may be added in future versions of this standard.

When processing information in a System Management Application Protocol (SMAP) PDU, the accepting SMAP-machine shall issue RORJapdu (corresponding to the service RO-REJECT-U) with "mistypedResult" parameter for the following conditions:

- enumerations not recognized
- unrecognized named numbers
- unrecognized named bits
- unrecognized tagged elements of sets, sequences, and choices

In addition, when new OBF data elements are added to this standard, they will be included in the additionalInformation attribute or parameter of a notification. This will cause new parameter templates to be defined. Objects which have additionalInformation as an attribute will not be subclassed, and T1M1 will not need to re-register the attribute or associated object templates. This is done to meet OBF requirements.

8.4 Functional Units and Services

8.4.1 Table of Functional Units, Services, and Objects

Functional units allow negotiation for use of various services in association (during association establishment). Table 1 lists the EAO functional units and corresponding services and object classes.

Table 1. LocationValidation Action Parameters

Functional Unit	Services	Object Classes
Inquiry	locationValidation action	eaoValidator
	serviceAvailability action	
	channelInformation action	eaoFacility
	PT-GET	

8.4.2 Service Definitions

The following services are defined in Recommendation X.730:

PT-GET

This section provides definitions for the following services:

locationValidation action

serviceAvailability action

channelInformation action

The definition of each EAO service in this standard includes a table which lists the parameters of its primitives. For a given primitive, the presence of each parameter is described by one of the following values:

M	The parameter is mandatory;
(=)	The value of the parameter is equal to the value of the parameter in the column to the left;
U	Use of the parameter is a service-user option;
–	The parameter is not present in the interaction;
C	The parameter is conditionally present (the conditions are described by the text that describes the parameter).

The service definitions in clause 7.2 are described using the service definition conventions specified in ITU-T Recommendation X.210.

8.4.2.1 LocationValidation Action Service

This service is used to allow a managing system to request an action on an eaoValidator. For a more detailed description of the action provided by the service, see the behaviour definition of locationValidation in section 6.4. This action uses the CMIS M-ACTION service. Table 2 provides the parameters for this action.

Table 2. LocationValidation Action Parameters

Parameter Name	Req/Ind	Rsp/Cnf
InvokeId	M	M(=)
Linked Identifier	-	C
Mode	M	-
Base Object Class	M	-
Base Object Instance	M	-
Scope	U	-
Filter	U	-
Managed Object Class	-	C
Managed Object Instance	-	C
Access Control	M	-
Action Information The action information for each kind of actions includes parameters that correspond to fields in the following action request syntax type: – LocationValidationRequest	M	-

Mandatory and optional fields for the types are shown in the syntax productions.		
Current Time	U	U
Action Reply The action reply for each kind of action includes parameters that correspond to fields in the following action reply syntax type: – LocationValidationReply Mandatory and optional fields for the type are shown in the syntax production.	-	M
Errors	-	C

8.4.2.2 ServiceAvailability Action Service

This service is used to allow a managing system to request an action on an eaoValidator. For a more detailed description of the action provided by the service, see the behaviour definition of serviceAvailability in section 6.4. This actions uses the CMIS M-ACTION service. Table 3 provides the parameters for this action

Table 3. Service Availability Action Parameters

Parameter Name	Req/Ind	Rsp/Cnf
InvokeId	M	M(=)
Linked Identifier	-	C
Mode	M	-
Base Object Class	M	-
Base Object Instance	M	-
Scope	U	-
Filter	U	-
Managed Object Class	-	C
Managed Object Instance	-	C
Access Control	M	-
Action Information The action information for each kind of actions includes parameters that correspond to fields in the following action request syntax type: – ServiceAvailabilityRequest Mandatory and optional fields for the types are shown	M	-

in the syntax productions.		
Current Time	U	U
Action Reply The action reply for each kind of action includes parameters that correspond to fields in the following action reply syntax type: – ServiceAvailabilityReply Mandatory and optional fields for the type are shown in the syntax production.	-	M
Errors	-	C

8.4.2.3 ChannelInformation Action Service

This service is used to allow a managing system to request an action on an eaoFacility. For a more detailed description of the action provided by the service, see the behaviour definition of channelInformation in section 6.4. This action uses the CMIS M-ACTION service. Table 4 provides the parameters for this action.

Table 4. ChannelInformation Action Parameters

Parameter Name	Req/Ind	Rsp/Cnf
InvokeId	M	M(=)
Linked Identifier	-	-
Mode	M	-
Base Object Class	M	-
Base Object Instance	M	-
Scope	-	-
Filter	U	-
Managed Object Class	-	M
Managed Object Instance	-	M
Access Control	-	M
Action Information The action information for each kind of actions includes parameters that correspond to fields in the following action request syntax type: – ChannelInfoRequest Mandatory and optional fields for the types are shown in the syntax productions.	M	-
Current Time	U	U
Action Reply	M	-

The action reply for each kind of action includes parameters that correspond to fields in the following action reply syntax type: – ChannelInfoReply Mandatory and optional fields for the type are shown in the syntax production.		
Errors	-	C

8.4.3 Negotiation of Functional Units

This specification assigns the following object identifier value

{iso(1) member-body(2) usa(840) ansi-t1-xxx-1996(x) functionalUnitPackage(1)}

as a value of the ASN.1 type FunctionalUnitPackagedId defined in CCITT Recommendation X701|ISO/IEC 10040 to use for negotiating the following functional units:

1 Inquiry

where the number identifies the bit position assigned to the functional unit, and the name refers to the functional unit as defined in Section 7 of this model.

8.5 Application Service Elements and Application Context

The System Management application context defined in CCITT Rec. X.701| ISO/IEC 10040 may be used.

8.6 Access Control

The second Access Control syntax definition and corresponding registered abstract syntax ({iso(1) member-body(2) usa(840) ansi t1-228-1996(10016) specificAccessControlAbstractSyntax(4) encryptionMethod(1)}) defined in T1.228 may be used for the access control parameter in CMIPUserInfo of A-ASSOCIATE requests and responses and CMIP operations request PDUs. Other approaches to security are not precluded.

8.7 Conformance

An implementation claiming to conform to this model shall comply with the requirements as defined in the following subclauses.

8.7.1 Static Conformance

The model requires the use of protocol requirements defined in T1.208 for session, presentation, and application layers of the OSI Reference Model. For CMISE, two profiles are defined in ISP 11183 Parts 1 and 3 (Basic Communications) and ISP 11183 Parts 1 and 2 (Enhanced Communications). Support of the Basic Communications profile is mandatory. Support of the Enhanced Communications profile allowing the use of scoping and filtering features is optional.

The system shall:

a) support the role of manager and agent or both, with respect to the functional units defined or referenced in this model,

b) support the transfer syntax derived from the encoding rules specified in CCITT Rec. X.209 and named {joint-iso-ccitt(2) asn1(1) basicEncoding(1)}, for the purpose of generating and/or interpreting the MAPDUs defined by the abstract data types defined in the section 6 of this model for the role supported in a) above, and

c) when acting in the agent role, for each functional unit for which conformance is claimed, support one or more instances of each object class listed in Table 1.

8.7.2 Dynamic Conformance

The system shall, in the role(s) and for the functional units for which conformance is claimed,

a) support the elements of procedure defined in:
   – CCITT Rec X.730 for the PT-GET service.

b) support the procedures for the services defined in section 7.2.

8.8 Mapping to OBF Requirements

The following sections map the data elements found in the OBF Requirements to the GDMO/ASN information model.

8.8.1 LocationInquiry

LocationInquiry	
OBF Data Element	**Object Model**
LIT	Implicit, based on operation in LocationInquiry interface
AN	Actions: validateCivicAddress retrieveByServiceId retrieveByLocationCode Defined in: NumberedAddress
STDIR	Actions: validateCivicAddress retrieveByServiceId retrieveByLocationCode Defined in: HouseNumberRangeAddress, NumberedAddress, UnnumberedAddress
STTYP	Actions: validateCivicAddress retrieveByServiceId retrieveByLocationCode Defined in: HouseNumberRangeAddress, NumberedAddress, UnnumberedAddress
STREET	Actions: validateCivicAddress retrieveByServiceId retrieveByLocationCode Defined in: HouseNumberRangeAddress, NumberedAddress, UnnumberedAddress
ROOM	Actions: validateCivicAddress

Location Inquiry		
OBF Data Element	**Object Model**	
		retrieveByServiceId retrieveByLocationCode
	Defined in:	HouseNumberRangeAddress, NumberedAddress, UnnumberedAddress
FLOOR	Actions:	validateCivicAddress retrieveByServiceId retrieveByLocationCode
	Defined in:	HouseNumberRangeAddress, NumberedAddress, UnnumberedAddress
BLDG	Actions:	validateCivicAddress retrieveByServiceId retrieveByLocationCode
	Defined in:	HouseNumberRangeAddress, NumberedAddress, UnnumberedAddress
CITY	Actions:	validateCivicAddress retrieveByServiceId retrieveByLocationCode
	Defined in:	HouseNumberRangeAddress, NumberedAddress, UnnumberedAddress, UnfieldedAddress
STATE	Actions:	validateCivicAddress retrieveByServiceId retrieveByLocationCode
	Defined in:	HouseNumberRangeAddress, NumberedAddress, UnnumberedAddress, UnfieldedAddress
CTRY	Actions:	validateCivicAddress retrieveByServiceId retrieveByLocationCode
	Defined in:	HouseNumberRangeAddress, NumberedAddress, UnnumberedAddress, UnfieldedAddress
POSTAL CODE	Actions:	validateCivicAddress retrieveByServiceId retrieveByLocationCode
	Defined in:	HouseNumberRangeAddress, NumberedAddress, UnnumberedAddress, UnfieldedAddress
LOCCODE	Operations:	retrieveByLocationCode
	Defined in:	LocationCode
ECCKT	Operations:	retrieveByServiceId
	Defined in:	serviceId

Location Inquiry	
OBF Data Element	**Object Model**
SWC	Operations: validateCivicAddress retrieveByServiceId retrieveByLocationCode Defined in: LocationCode
LSO	Operations: validateCivicAddress retrieveByServiceId retrieveByLocationCode Defined in: LocationCode
IRM	Actions: validateCivicAddress retrieveByServiceId retrieveByLocationCode Defined in: additionalInfo, additionalInformation, additionalText

8.8.2 Service Availability

Service Availability	
OBF Data Element	**Object Model**
NC, NCI, SECNCI, SPEC	Actions: retrieveAvailability, Defined in: ServiceCodeDefinition
QTY	Actions: retrieveAvailability, Defined in: Quantity
LOC 1, LOC 2	Actions: retrieveAvailability, Defined in: locations
DDD	Actions: retrieveAvailability, Defined in: availabilityDate
PIU	Actions: retrieveAvailability, Defined in: percentInterstateUsage
IRI	Actions: retrieveAvailability, Defined in: Implicit by the return value
IRM	Actions: retrieveAvailability Defined in: additionalInfo, additionalInformation, additionalText

8.8.3 CFA Inquiry

CFA Inquiry	
OBF Data Element	**Object Model**
FAC DESG, FAC TYP, LOC A, LOC Z	Actions: retrieveChannelInformation Defined in: serviceId
CHN	Actions: retrieveChannelInformation Defined in: ChannelNumbersList
ECCKT	Actions: retrieveChannelInformation Defined in: serviceId
STATUS	Actions: retrieveChannelInformation Defined in: ChannelStatus

CFA Inquiry	
OBF Data Element	**Object Model**
CKR	Actions: retrieveChannelInformation Defined in: serviceAliasList
PA	Operations: retrieveChannelInformation Defined in: ServiceActivity
DD	Operations: retrieveChannelInformation Defined in: dueDate
PON	Operations: retrieveChannelInformation Defined in: purchaseOrderNumber
IRI	Operations: retrieveChannelInformation Defined in: Implicit by the return value
IRM	Operations: retrieveChannelInformation Defined in: additionalInfo, additionalInformation, additionalText

Annex A: Deprecated GDMO/ASN (Normative)

Below is GDMO/ASN that has been deprecated from the original version of this standard. This GDMO/ASN may become part of the standard for EAO ordering functions (OBF functions G4.0 through G8.0).

A.1 Managed Object Classes

A.1.1 cnmServiceR1

A.1.1.1 Description

The cnmServiceR1 is a revision of the cnmService managed object class defined in T1.227. The revision was necessary to accommodate more descriptive address information required by OBF. The only difference between this managed object class and that defined in T1.227 is the ASN syntax for the serviceLocationList.

A.1.1.2 Template

```
cnmServiceR1 MANAGED OBJECT CLASS
  DERIVED FROM "T1.240":service;
  CHARACTERIZED BY
    cnmServiceR1Pkg PACKAGE
      BEHAVIOUR
        cnmServiceR1Behaviour BEHAVIOUR
        DEFINED AS
        "This is the superclass for the cnmAccessService and
        eaoFacility object classes.";;
      ATTRIBUTES
          -- 2 mandatory attributes inherited from service:
          -- serviceID     GET,
          -- serviceType    GET-REPLACE,
          "T1.227":serviceDescription   GET,
          serviceLocationListR1  GET;;;
  CONDITIONAL PACKAGES

    "T1.227":csServiceAliasPkg
      PRESENT IF "an instance supports it",
     "T1.227":csServiceProfileObjectPtrPkg
          PRESENT IF "an instance supports it",
     "T1.227":csTroubleReportFormatObjectPtrPkg
          PRESENT IF "an instance supports it";

REGISTERED AS { eaoObjectClass  1};
```

A.1.2 cnmAccessService

A.1.2.1 Description

The cnmAccessService managed object class is an abstract base class which contains behaviour and attributes which are common to all access services.

A.1.2.2 Template

```
cnmAccessService MANAGED OBJECT CLASS
  DERIVED FROM  cnmServiceR1;
  CHARACTERIZED BY
    cnmAccessServicePkg PACKAGE
      BEHAVIOUR
        cnmAccessServiceBehaviour BEHAVIOUR
        DEFINED AS
        "cnmAccessService is an abstract base class which contains
        behaviour common to all access service classes.";;
      ATTRIBUTES
        networkChannelCode    GET,
        networkChannelInterfaceCode   GET;;;
  CONDITIONAL PACKAGES
     billingAccountNumberPkg
      PRESENT IF "an instance supports it",
    equipmentDesignationPkg
      PRESENT IF "an instance supports it",
    facilityPointerPkg
     PRESENT IF "this service is a channel of a facility",
    fiberNetworkIdPkg
         PRESENT IF "an instance supports it",
    highVoltageProtectionPkg
      PRESENT IF "an instance supports it",
    multipleProviderPkg
     PRESENT IF "this service spans multiple providers",
    percentInterstateUsagePkg
      PRESENT IF "an instance supports it",
    relayRackPkg
      PRESENT IF "an instance supports it",
    routeMilesPkg
      PRESENT IF "an instance supports it",
    secondaryNetworkChannelInterfaceCodePkg
      PRESENT IF "an instance supports it",
    secondaryTransmissionLevelPkg
      PRESENT IF "an instance supports it",
    serviceProductEnhancementCodePkg
      PRESENT IF "an instance supports it",
    specialRoutingPkg
      PRESENT IF "an instance supports it",
         tspPriorityR1Pkg
      PRESENT IF "an instance supports it",
    transferFeaturePkg
      PRESENT IF "an instance supports it",
    transmissionLevelPkg
```

```
      PRESENT IF "an instance supports it",
    unitNumberPkg
      PRESENT IF "an instance supports it";

REGISTERED AS { eaoObjectClass  2};
```

A.1.3 eaoFacility

A.1.3.1 Description

The eaoFacility managed object class represents a service which is multiplexed into multiple channels which are used to carry other lower bandwidth services.

A.1.3.2 Template

```
eaoFacility MANAGED OBJECT CLASS
  DERIVED FROM  cnmAccessService;
  CHARACTERIZED BY
       eaoFacilityPkg PACKAGE
           BEHAVIOUR
            eaoFacilityBehaviour BEHAVIOUR
           DEFINED AS
            "The eaoFacility represents the CLF and is used to derive
            specific types like DS1 and DS3. In a CFA inquiry, an
            eaoFacility object instance contains the channel
            information of CLF.  If a channel is busy, the serviceID
            (ECCKT) is to be provided in the reply. The eaoFacility is
            subclassed from the cnmAccessService object class. For
            additional behaviour, please refer to section 6.1.3.";;
           ATTRIBUTES
            channelList       GET;
           ACTIONS
                channelInformation        channelInformationError;;;

REGISTERED AS { eaoObjectClass  3};
```

A.1.4 eaoValidator

A.1.4.1 Description

The eaoValidator managed object class allows the service availability and address validation inquiry functions to be performed with a single CMIP request. Without this class, multiple CMIP requests and scoping and filtering would have to be used, thus complicating the object model and its implementation.

A.1.4.2 Template

```
eaoValidator  MANAGED OBJECT CLASS
  DERIVED FROM "Rec. X.721|ISO/IEC 10165-2": top;
  CHARACTERIZED BY
    eaoValidatorPkg  PACKAGE
     BEHAVIOUR
         eaoValidatorBehaviour BEHAVIOUR
         DEFINED AS
         "The eaoValidator is an object class designed to facilitate
         location and service availability inquiries. A single instance
```

```
        may support multiple inquiries. PercentInterstateUsage is
        mandatory for access services. For additional behaviour please
        refer to sections 6.1.1 and 6.1.2";;
    ATTRIBUTES
        validationId            GET;
    ACTIONS
        locationValidation          locationValidationError
                                    maxAlternativesExceeded,
        serviceAvailability         serviceAvailabilityError
                                    maxAlternativesExceeded;
    NOTIFICATIONS
        "Rec. X.721 | ISO/IEC 10165-2": objectCreation,
        "Rec. X.721 | ISO/IEC 10165-2": objectDeletion;;;

REGISTERED AS {eaoObjectClass 4};
```

A.2 Packages

The following packages have been defined for use in characterizing object classes.

A.2.1 billingAccountNumberPkg

```
billingAccountNumberPkg PACKAGE
  ATTRIBUTES
    billingAccountNumber   GET SET-BY-CREATE;
REGISTERED AS {eaoPackage  1};
```

A.2.2 equipmentDesignationPkg

```
equipmentDesignationPkg PACKAGE
  ATTRIBUTES
    equipmentDesignation                GET;
REGISTERED AS {eaoPackage  2};
```

A.2.3 facilityPointerPkg

```
facilityPointerPkg    PACKAGE
  ATTRIBUTES
    facilityPointer        GET SET-BY-CREATE;
REGISTERED AS {eaoPackage  3};
```

A.2.4 fiberNetworkIdPkg

```
fiberNetworkIdPkg  PACKAGE
  ATTRIBUTES
    fiberNetworkId    GET SET-BY-CREATE;
REGISTERED AS {eaoPackage  4};
```

A.2.5 highVoltageProtectionPkg

```
highVoltageProtectionPkg PACKAGE
```

```
  ATTRIBUTES
    highVoltageProtection   GET SET-BY-CREATE;
REGISTERED AS {eaoPackage  5};
```

A.2.6 multipleProviderPkg

```
multipleProviderPkg PACKAGE
  ATTRIBUTES
    multipleProvider          GET SET-BY-CREATE;
REGISTERED AS {eaoPackage  6};
```

A.2.7 percentInterstateUsagePkg

```
percentInterstateUsagePkg  PACKAGE
  ATTRIBUTES
    percentInterstateUsage   GET-REPLACE;
REGISTERED AS {eaoPackage  7};
```

A.2.8 relayRackPkg

```
relayRackPkg PACKAGE
  ATTRIBUTES
    relayRack          GET;
REGISTERED AS {eaoPackage  8};
```

A.2.9 routeMilesPkg

```
routeMilesPkg PACKAGE
  ATTRIBUTES
    routeMiles   GET;
REGISTERED AS {eaoPackage  9};
```

A.2.10 serviceProductEnhancementCodePkg

```
serviceProductEnhancementCodePkg  PACKAGE
  ATTRIBUTES
    serviceProductEnhancementCode  GET-REPLACE;
REGISTERED AS {eaoPackage  10};
```

A.2.11 specialRoutingPkg

```
specialRoutingPkg  PACKAGE
  ATTRIBUTES
    specialRouting     GET-REPLACE;
REGISTERED AS {eaoPackage  11};
```

A.2.12 secondaryNetworkChannelInterfaceCodePkg

```
secondaryNetworkChannelInterfaceCodePkg  PACKAGE
  ATTRIBUTES
    secondaryNetworkChannelInterfaceCode     GET-REPLACE;
REGISTERED AS {eaoPackage  12};
```

A.2.13 secondaryTransmissionLevelPkg

```
secondaryTransmissionLevelPkg  PACKAGE
  ATTRIBUTES
    secondaryTransmissionLevel GET-REPLACE;
REGISTERED AS {eaoPackage  13};
```

A.2.14 transmissionLevelPkg

```
transmissionLevelPkg  PACKAGE
  ATTRIBUTES
    transmissionLevel GET-REPLACE;
REGISTERED AS {eaoPackage  14};
```

A.2.15 transferFeaturePkg

```
transferFeaturePkg  PACKAGE
  ATTRIBUTES
    transferFeature GET-REPLACE;
REGISTERED AS {eaoPackage  15};
```

A.2.16 tspPriorityR1Pkg

```
tspPriorityR1Pkg  PACKAGE
  ATTRIBUTES
    "T1.227":tspPriority GET-REPLACE;
REGISTERED AS {eaoPackage  16};
```

A.2.17 unitNumberPkg

```
unitNumberPkg PACKAGE
  ATTRIBUTES
    unitNumber   GET;
REGISTERED AS {eaoPackage  17};
```

A.3 Actions

A.3.1 locationValidation

```
locationValidation   ACTION
      BEHAVIOUR
     locationValidationBehaviour  BEHAVIOUR
     DEFINED AS
     "The behaviour is defined in section 6.1.1.";;
      MODE CONFIRMED;
      WITH INFORMATION SYNTAX EAO.LocationValidationRequest;
      WITH REPLY SYNTAX EAO.LocationValidationReply;
REGISTERED AS {eaoAction   1};
```

A.3.2 serviceAvailability

```
serviceAvailability   ACTION
      BEHAVIOUR
     serviceAvailabilityBehaviour   BEHAVIOUR
     DEFINED AS
```

```
      "The behaviour is defined in section 6.1.2.";;
       MODE CONFIRMED;
       WITH INFORMATION SYNTAX EAO.ServiceAvailabilityRequest ;
       WITH REPLY SYNTAX EAO.ServiceAvailabilityReply;
REGISTERED AS {eaoAction   2};
```

A.3.3 channelInformation

```
channelInformation ACTION
   BEHAVIOUR
      channelInformationBehaviour  BEHAVIOUR
      DEFINED AS
      "The behaviour is defined in section 6.1.3.";;
   MODE CONFIRMED;
   WITH INFORMATION SYNTAX EAO.ChannelInfoRequest;
   WITH REPLY SYNTAX EAO.ChannelInfoReply;
REGISTERED AS {eaoAction   3};
```

A.4 Notifications

A.4.1 Object Creation

This notification is used to report the creation of a managed object if defined in the managed object class specification. It is defined in Recommendation X.721.

A.4.2 Object Deletion

This notification is used to report the deletion of a managed object if defined in the managed object class specification. It is defined in Recommendation X.721.

A.5 Parameters

A.5.1 locationValidationError

```
locationValidationError PARAMETER
   CONTEXT SPECIFIC-ERROR;
       WITH SYNTAX EAO.LocationValidationError;
       BEHAVIOUR
      locationValidationErrorBehaviour  BEHAVIOUR
      DEFINED AS
        "This parameter is used when an error occurs in processing a
        locationValidation action request for an eaoValidator object
        instance. See section 6.2.1 for more details.";;
REGISTERED AS {eaoParameter    1};
```

A.5.2 serviceAvailabilityError

```
serviceAvailabilityError PARAMETER
   CONTEXT SPECIFIC-ERROR;
   WITH SYNTAX EAO.ServiceAvailabilityError;
```

```
   BEHAVIOUR
      serviceAvailabilityErrorBehaviour  BEHAVIOUR
      DEFINED AS
         " This parameter is used when an error occurs in processing a
         serviceAvailability action request for an eaoValidator object
         instance.  See section 6.2.2 for more details. ";;
REGISTERED AS {eaoParameter    2};
```

A.5.3 channelInformationError

```
channelInformationError PARAMETER
   CONTEXT SPECIFIC-ERROR;
   WITH SYNTAX EAO.ChannelInformationError;
   BEHAVIOUR
      channelInformationErrorBehaviour  BEHAVIOUR
      DEFINED AS
         "This parameter is used when an error occurs in processing a
         channelInfomation action request for an eaoFacility object
         instance. See section 6.2.1 for more details.";;
REGISTERED AS {eaoParameter    3};
```

A.5.4 maxAlternativesExceeded

```
maxAlternativesExceeded PARAMETER
   CONTEXT SPECIFIC-ERROR;
   WITH SYNTAX EAO.MaxAlternativesExceeded;
   BEHAVIOUR
      maxAlternativesExceededBehaviour BEHAVIOUR
      DEFINED AS
         "This parameter is used when an implementor has set a limit to
         the number of alternatives that can be returned on a
         locationValidation or serviceAvailability action.";;
REGISTERED AS {eaoParameter    4};
```

A.6 Attributes

A.6.1 billingAccountNumber

```
billingAccountNumber   ATTRIBUTE
   WITH ATTRIBUTE SYNTAX  EAO.BillingAccountNumber;
   MATCHES FOR EQUALITY, SUBSTRINGS;
   BEHAVIOUR
       billingAccountNumberBehaviour BEHAVIOUR
       DEFINED AS
       "Identifies the billing account to which the recurring and
      non-recurring charges for this request will be billed";;
REGISTERED AS {eaoAttribute    1};
```

A.6.2 channelList

```
channelList ATTRIBUTE
   WITH ATTRIBUTE SYNTAX  EAO.ChannelList;
   MATCHES FOR EQUALITY, SET-INTERSECTION, SET-COMPARISON;
   BEHAVIOUR
      channelListBehaviour BEHAVIOUR
      DEFINED AS
      "Identifies the facility's channels and the services assigned to
      those channels (updated internally as the service orders
      complete.)";;
REGISTERED AS {eaoAttribute    2};
```

A.6.3 equipmentDesignation

```
equipmentDesignation   ATTRIBUTE
   WITH ATTRIBUTE SYNTAX  EAO.EquipmentDesignation;
   MATCHES FOR EQUALITY, SUBSTRINGS;
   BEHAVIOUR
       equipmentDesignationBehaviour BEHAVIOUR
       DEFINED AS
      "A code identifying a piece of equipment at the customer
      location";;
REGISTERED AS {eaoAttribute    3};
```

A.6.4 facilityPointer

```
facilityPointer   ATTRIBUTE
   WITH ATTRIBUTE SYNTAX  EAO.FacilityPointer;
   MATCHES FOR EQUALITY;
   BEHAVIOUR
       facilityPointerBehaviour BEHAVIOUR
       DEFINED AS
       "Identifies the facility that this service is provisioned on";;
REGISTERED AS {eaoAttribute    4};
```

A.6.5 fiberNetworkId

```
fiberNetworkId   ATTRIBUTE
   WITH ATTRIBUTE SYNTAX  EAO.FiberNetworkId;
   MATCHES FOR EQUALITY, SUBSTRINGS;
   BEHAVIOUR
       fiberNetworkIdBehaviour BEHAVIOUR
       DEFINED AS
      "Identifies the fiber network that this service is provisioned
      on.";;
REGISTERED AS {eaoAttribute    5};
```

A.6.6 highVoltageProtection

```
highVoltageProtection    ATTRIBUTE
   WITH ATTRIBUTE SYNTAX  EAO.HighVoltageProtection;
```

```
   MATCHES FOR EQUALITY;
   BEHAVIOUR
      highVoltageProtectionBehaviour BEHAVIOUR
      DEFINED AS
     "Indicates the requirements for high voltage protection at a point
     of terminiation";;
REGISTERED AS {eaoAttribute    6};
```

A.6.7 multipleProvider

```
multipleProvider ATTRIBUTE
   WITH ATTRIBUTE SYNTAX  EAO.MultipleProvider;
   MATCHES FOR EQUALITY, SUBSTRINGS;
   BEHAVIOUR
            multipleProviderBehaviour BEHAVIOUR
            DEFINED AS
     "Identifies the ICSC code of the Access Service Coordination -
     Exchange Company (ASC-EC) whenever an access service passes through
     more than one access provider's territory";;
REGISTERED AS {eaoAttribute    7};
```

A.6.8 networkChannelCode

```
networkChannelCode    ATTRIBUTE
   WITH ATTRIBUTE SYNTAX  EAO.NCCode;
   MATCHES FOR EQUALITY, SUBSTRINGS;
   BEHAVIOUR
      networkChannelCodeBehaviour BEHAVIOUR
      DEFINED AS
     "Identifies the characteristics for the service channel.";;
REGISTERED AS {eaoAttribute    8};
```

A.6.9 networkChannelInterfaceCode

```
networkChannelInterfaceCode    ATTRIBUTE
   WITH ATTRIBUTE SYNTAX  EAO.NCICode;
   MATCHES FOR EQUALITY, SUBSTRINGS;
   BEHAVIOUR
      networkChannelInterfaceCodeBehaviour BEHAVIOUR
      DEFINED AS
     "Identifies the electrical conditions on the service at the access
     carrier terminal location.";;
REGISTERED AS {eaoAttribute    9};
```

A.6.10 percentInterstateUsage

```
percentInterstateUsage    ATTRIBUTE
   WITH ATTRIBUTE SYNTAX  EAO.PercentInterstateUsage;
   MATCHES FOR EQUALITY, ORDERING;
   BEHAVIOUR
     percentInterstateUsageBehaviour BEHAVIOUR
```

```
      DEFINED AS
     "Identifies the expected interstate usage for the service. Matches
     for ordering is only applicable to the integer choice";;
REGISTERED AS {eaoAttribute    10};
```

A.6.11 relayRack

```
relayRack    ATTRIBUTE
  WITH ATTRIBUTE SYNTAX  EAO.RelayRack;
  MATCHES FOR EQUALITY, SUBSTRINGS;
  BEHAVIOUR
      relayRackBehaviour BEHAVIOUR
      DEFINED AS
     "Identifies the floor, aisle, and bay/cabinet where the equipment
     indicated in the equipmentDesignator attribute is located.";;
REGISTERED AS {eaoAttribute    11};
```

A.6.12 routeMiles

```
routeMiles   ATTRIBUTE
  WITH ATTRIBUTE SYNTAX   EAO.RouteMiles;
  MATCHES FOR EQUALITY, ORDERING;
  BEHAVIOUR
      routeMilesBehaviour BEHAVIOUR
      DEFINED AS
     "The total length of all facilities used to provide this
     service.";;
REGISTERED AS {eaoAttribute    12};
```

A.6.13 secondaryNetworkChannelInterfaceCode

```
secondaryNetworkChannelInterfaceCode ATTRIBUTE
  WITH ATTRIBUTE SYNTAX   EAO.NCICode;
  MATCHES FOR EQUALITY, ORDERING;
 REGISTERED AS {eaoAttribute    13};
```

A.6.14 secondaryTransmissionLevel

```
secondaryTransmissionLevel ATTRIBUTE
  WITH ATTRIBUTE SYNTAX   EAO.TransmissionLevel;
  MATCHES FOR EQUALITY, ORDERING;
 REGISTERED AS {eaoAttribute    14};
```

A.6.15 serviceLocationListR1

```
serviceLocationListR1  ATTRIBUTE
```

```
   WITH ATTRIBUTE SYNTAX   EAO.ServiceLocationListR1;
   MATCHES FOR EQUALITY, SET-INTERSECTION, SET-COMPARISON;
   BEHAVIOUR
       serviceLocationListR1Behaviour BEHAVIOUR
       DEFINED AS
       "Identifies the various locations that the service intersects.";;
REGISTERED AS {eaoAttribute    15};
```

A.6.16 serviceProductEnhancementCode

```
serviceProductEnhancementCode   ATTRIBUTE
   WITH ATTRIBUTE SYNTAX   EAO.SPECCode;
   MATCHES FOR EQUALITY,SUBSTRINGS;
   BEHAVIOUR
       serviceProductEnhancementCodeBehaviour BEHAVIOUR
       DEFINED AS
       "Identifies the specific product or service offering";;
REGISTERED AS {eaoAttribute    16};
```

A.6.17 specialRouting

```
specialRouting    ATTRIBUTE
   WITH ATTRIBUTE SYNTAX   EAO.SpecialRouting;
   MATCHES FOR EQUALITY;
   BEHAVIOUR
       specialRoutingBehaviour BEHAVIOUR
       DEFINED AS
       "Identifies the type of special routing needed for this
   service.";;
REGISTERED AS {eaoAttribute    17};
```

A.6.18 transferFeature

```
transferFeature ATTRIBUTE
   WITH ATTRIBUTE SYNTAX  EAO.TransferFeature;
   MATCHES FOR EQUALITY;
   BEHAVIOUR
       transferFeatureBehaviour BEHAVIOUR
       DEFINED AS
      "Identifies the transfer feature indicator for the transfer
      relay.";;
REGISTERED AS {eaoAttribute    18};
```

A.6.19 transmissionLevel

```
transmissionLevel    ATTRIBUTE
   WITH ATTRIBUTE SYNTAX   EAO.TransmissionLevel;
   MATCHES FOR EQUALITY, SUBSTRINGS;
   BEHAVIOUR
       transmissionLevelBehaviour BEHAVIOUR
       DEFINED AS
```

```
      "The transmission level for the service.";;
REGISTERED AS {eaoAttribute    19};
```

A.6.20 unitNumber

```
unitNumber   ATTRIBUTE
   WITH ATTRIBUTE SYNTAX   EAO.UnitNumber;
   MATCHES FOR EQUALITY, SUBSTRINGS;
   BEHAVIOUR
      unitNumberBehaviour BEHAVIOUR
      DEFINED AS
     "The number assigned to a panel, shelf or case within the
     bay/cabinet indicated in the relayRack attribute.";;
REGISTERED AS {eaoAttribute    20};
```

A.6.21 validationId

```
validationId   ATTRIBUTE
   WITH ATTRIBUTE SYNTAX   ASN1DefinedTypesModule.NameType;
   MATCHES FOR EQUALITY;
   BEHAVIOUR "Recommendation X.721:1992":rDNIdBehaviour;
     -- The above behaviour is defined as part of discriminatorId
     -- in Rec. X.721
REGISTERED AS {eaoAttribute    21};
```

A.7 Name Bindings

A.7.1 cnmServiceR1-account

```
cnmServiceR1-account NAME BINDING
   SUBORDINATE OBJECT CLASS cnmServiceR1 AND SUBCLASSES;
   NAMED BY
   SUPERIOR OBJECT CLASS "T1.227":account;
   WITH ATTRIBUTE "T1.240":serviceID;
REGISTERED AS {eaoNameBinding 1};
```

A.7.2 eaoValidator-account

```
eaoValidator-account NAME BINDING
   SUBORDINATE OBJECT CLASS eaoValidator AND SUBCLASSES;
   NAMED BY
   SUPERIOR OBJECT CLASS "T1.227":account;
   WITH ATTRIBUTE validationId;
REGISTERED AS {eaoNameBinding 2};
```

A.7.3 eaoValidator-network

```
eaoValidator-network NAME BINDING
   SUBORDINATE OBJECT CLASS eaoValidator AND SUBCLASSES;
   NAMED BY
   SUPERIOR OBJECT CLASS "Rec. M.3100":network;
   WITH ATTRIBUTE validationId;
```

```
REGISTERED AS {eaoNameBinding 3};
```

A.8 Extensibility Rules

In accordance with ISO 8824-1 Amendment 1 on extensibility rules, the productions that are of extensible types are to be indicated by including the following in their type descriptions ...

The following types will be indicated as being extensible:

- ENUMERATED
- named INTEGER
- named BIT STRING
- SET
- SEQUENCE
- CHOICE

Under the rules of extensibility new enumerations (for ENUMERATED types), new bit name assignments (for named BIT STRING types), new named numbers (for named INTEGER types), and new elements (for SET, SEQUENCE, and CHOICE types) may be added in future versions of this standard.

When processing information in a System Management Application Protocol (SMAP) PDU, the accepting SMAP-machine shall issue RORJapdu (corresponding to the service RO-REJECT-U) with "mistypedResult" parameter for the following conditions:

- enumerations not recognized
- unrecognized named numbers
- unrecognized named bits.
- unrecognized tagged elements of sets, sequences, and choices

In addition, when new OBF data elements are added to this standard, they will be included in the additionalInformation attribute or parameter of a notification. This will cause new parameter templates to be defined. Objects which have additionalInformation as an attribute will not be subclassed, and T1M1 will not need to re-register the attribute or associated object templates. This is done to meet OBF requirements.

A.9 Supporting Productions

```
EAO {iso(1) member-body(2) usa(840) ansi-t1-256-1996(x)  eao(0)}

DEFINITIONS IMPLICIT TAGS ::=  BEGIN

-- EXPORTS Everything

IMPORTS

ObjectInstance
   FROM CMIP-1 {joint-iso-ccitt(2) ms(9) cmip(1) modules(0) protocol(3)}
```

```
ChannelNumber, NameType, ObjectList
   FROM ASN1DefinedTypesModule  { ccitt recommendation m(13) gnm(3100)
   informationModel(0) asn1Modules(2)  asn1DefinedTypesModule(0) }

ServiceAliasList
   FROM GNMTA {iso(1) member-body(2) usa(840) ansi-t1-227-1992(10015) trGNM(0)
   gnmta(0)} ;

eaoObjectClass OBJECT IDENTIFIER::={iso(1) member-body(2) usa(840) ansi-t1-256-
1996(x) eao(0) objectClass(3)}

eaoPackage OBJECT IDENTIFIER::={iso(1) member-body(2) usa(840) ansi-t1-256-
1996(x) eao(0) package(4)}

eaoParameter OBJECT IDENTIFIER::={iso(1) member-body(2) usa(840) ansi-t1-256-
1996(x) eao(0) parameter(5)}

eaoNameBinding OBJECT IDENTIFIER::={iso(1) member-body(2) usa(840) ansi-t1-256-
1996(x) eao(0) nameBinding(6)}

eaoAttribute OBJECT IDENTIFIER::={iso(1) member-body(2) usa(840) ansi-t1-256-
1996(x) eao(0) attribute(7)}

eaoBehaviour OBJECT IDENTIFIER::={iso(1) member-body(2) usa(840) ansi-t1-256-
1996(x) eao(0) behaviour(8)}

eaoAction OBJECT IDENTIFIER::={iso(1) member-body(2) usa(840) ansi-t1-256-
1996(x) eao(0) action(9)}

eaoNotification OBJECT IDENTIFIER::={iso(1) member-body(2) usa(840) ansi-t1-256-
1996(x) eao(0) notification(10)}

Address  ::= SEQUENCE {
   number            [0]          AddrNumber
       OPTIONAL,
   streetDirection   [1]          AddrStreetDirection
       OPTIONAL,
   street            [2]          AddrStreet,
   streetType        [3]          AddrStreetType
       OPTIONAL,
   mailStop          [4]          AddrMailStop
   OPTIONAL,
   room              [5]          AddrRoom
       OPTIONAL,
   floor             [6]          AddrFloor
       OPTIONAL,
   building          [7]          AddrBuilding
       OPTIONAL,
   city              [8]          AddrCity,
   stateOrProvince   [9]          AddrStateOrProvince,
```

```
   country          [10]          AddrCountry
       OPTIONAL,
   postalCode       [11]          AddrPostalCode
       ...
}

AddrNumber ::=  PrintableString(SIZE(0..64))

AddrMailStop ::=  PrintableString(SIZE(0..64))

AddrStreetDirection  ::= PrintableString(SIZE(0..64))

AddrStreet ::= PrintableString(SIZE(0..64))

AddrStreetType  ::= PrintableString(SIZE(0..64))

AddrRoom  ::= PrintableString(SIZE(0..64))

AddrFloor  ::= PrintableString(SIZE(0..64))

AddrBuilding  ::= PrintableString(SIZE(0..64))

AddrCity  ::= PrintableString(SIZE(0..64))

AddrStateOrProvince  ::= PrintableString(SIZE(0..64))

AddrCountry  ::= PrintableString(SIZE(0..64))

AddrPostalCode  ::= PrintableString(SIZE(0..64))

AssignedChannelInfo ::=   SEQUENCE{
  serviceId                [0]    ConfidentialServiceId,
  serviceAliasList         [1]    ConfidentialServiceAliasList
       OPTIONAL
       ...
}

BillingAccountNumber  ::=  PrintableString

ChannelInfo ::= SEQUENCE {
   channels         [0]          SET OF ChannelNumbers,
   info             [1]          ChannelStatus
       ...
}

ChannelInfoReply ::=  SEQUENCE
{
-- in the action reply, ranges or sets of channels tied to a single
-- channel status are only allowed for SPARE or ASSIGNED only.
```

```
    info                    [0]    SET OF ChannelInfo,
    additionalText          [1]    PrintableString     OPTIONAL
-- additionalText in ChannelInfoReply is used to convey error
-- messages or other messages concerning the CFA inquiry.
        ...
}

ChannelInfoRequest ::= SET OF ChannelNumbers

ChannelInformationError ::=  SEQUENCE {
   reason                   [0]    ErrorNotFoundOrAccessDenied,
   additionalText           [1]    PrintableString            OPTIONAL
   ...
}

ChannelList ::= SET OF SEQUENCE {
   channelNumber            ChannelNumber,
   serviceId                ObjectInstance
        ...
}



ChannelNumbers ::= CHOICE {
   single            ChannelNumber,
   range             ChannelRange
}

ChannelRange ::= SEQUENCE {
   from              [0]    ChannelNumber,
   to                [1]    ChannelNumber
}

ChannelStatus ::= CHOICE {
   spare                   [0]    NULL,
   assigned                [1]    AssignedChannelInfo,
   pending                 [2]    PendingChannelInfo,
   channelAccessDenied     [3]    NULL,
   invalid                 [4]    NULL
        ...
}

CivicAddress ::=  CHOICE {
   descriptiveAddress      [0]    UnnumberedAddress,
   numberedAddress         [1]    NumberedAddress,
   unnumberedAddress       [2]    UnnumberedAddress
}

ConfidentialCivicAddress ::= CHOICE {
```

```
   civicAddress                    CivicAddress,
   confidentialCivicAddress        ConfidentialString
        ...
}

ConfidentialEndUser ::= CHOICE {
       endUser                     EndUser,
       confidentialEndUser         ConfidentialString
       ...
}

ConfidentialLocCode ::= CHOICE {
   locCode                         LocCode,
   confidentialLocCode             ConfidentialString
        ...
}

ConfidentialPurchaseOrderNumber ::= CHOICE {
       purchaseOrderNumber
       PurchaseOrderNumber,
       confidentialPurchaseOrderNumber          ConfidentialString
       ...
}

ConfidentialServiceAliasList ::= CHOICE {
   clearString                     [0]          ServiceAliasList,
   confidentialString              [1]          ConfidentialString
        ...
}

ConfidentialServiceId ::= CHOICE {
   serviceId                       NameType,
   confidentialServicdId           ConfidentialString
        ...
}

ConfidentialString ::= OCTET STRING

Date ::= GeneralizedTime

EaoFacilityPointer ::= ObjectInstance

EndUser       ::=    SEQUENCE {
   name                      PrintableString,
   address                   CivicAddress
        ...
}

ErrorNotFound  ::=  INTEGER {
```

```
   notFound  (0)
       ...
}

ErrorNotFoundOrAccessDenied ::= INTEGER   {
   notFound          (0),
   accessDenied      (1)
       ...
}

EquipmentDesignation ::= PrintableString

FacilityPointer ::= ObjectInstance

FiberNetworkId  ::= PrintableString(SIZE(1..13))

HighVoltageProtection  ::=  ENUMERATED {
       required            (0),
       remove              (1)
       ...
}

LocAddressFoundReply::= SEQUENCE {
   civicAddress                  [0]    ConfidentialCivicAddress
       OPTIONAL,
   servingWireCenter             [1]    LocCode
       OPTIONAL,
   localServingOffice            [2]    LocalServingOffice
       OPTIONAL,
   additionalText                [3]    PrintableString
       OPTIONAL
       ...
}

LocCode  ::= PrintableString (SIZE (8|11))

LocalServingOffice  ::= PrintableString(SIZE (6))

Location ::= CHOICE {
       locCode             [0]    LocCode,
       address             [1]    CivicAddress
       ...
}

LocationInfo::=   CHOICE {
   accessCustomerTerminationLoc          [0]    LocCode,
   primaryLoc                    [1]    Location,
   secondaryLoc                  [2]    Location,
   servingWireCenter             [3]    LocCode,
   muxLoc                        [4]    LocCode,
```

```
   localServingOffice              [5]    LocalServingOffice,
   endOffice                       [6]    LocCode,
   endUser                         [7]    ConfidentialEndUser,
   altServingWireCenter            [8]    LocCode
       ...
}

LocationValidationError ::= CHOICE {
   civicAddressErrorReason         [0]    ErrorNotFound,
   serviceIdErrorReason            [1]    ErrorNotFoundOrAccessDenied,
   locationCodeErrorReason         [2]    ErrorNotFound
       ...
}

LocationValidationReply ::= CHOICE {
   civicAddressReply               [0]    SET OF LocAddressFoundReply,
   serviceIdReply                  [1]    LocAddressFoundReply,
   locationCodeReply               [2]    LocAddressFoundReply
       ...
}

LocationValidationRequest ::= CHOICE {
   civicAddress                    [0]    ConfidentialCivicAddress,
   serviceId                       [1]    ConfidentialServiceId,
   locationCode                    [2]    ConfidentialLocCode
       ...
}

Locations ::= SEQUENCE
{
       location1                   [0]          ConfidentialLocCode,
       location2                   [1]          ConfidentialLocCode
       ...
}

MaxAlternativesExceeded ::= PrintableString

MultipleProvider ::= PrintableString

NCCode  ::= PrintableString (SIZE (4))

NCICode  ::= PrintableString (SIZE (5..12))

NumberedAddress  ::= Address (
   WITH COMPONENTS {
              ... ,
              number         PRESENT
   }
)
```

```
PendingChannelInfo ::= SEQUENCE {
   activity                        [0]    ServiceActivity,
   dueDate                         [1]    Date,
   serviceId                       [2]    ConfidentialServiceId
       OPTIONAL,
   serviceAliasList                [3]    ConfidentialServiceAliasList
       OPTIONAL,
   purchaseOrderNumber             [4]    ConfidentialPurchaseOrderNumber
       OPTIONAL
       ...
}

PercentInterstateUsage ::= CHOICE {
   amount                          [0]    INTEGER(0..100),
   letterOnFile                    [1]    NULL
       ...
}

PointOfTermination  ::= PrintableString

PurchaseOrderNumber ::= PrintableString

Quantity ::= INTEGER(1..MAX)

RelayRack ::= PrintableString

RouteMiles ::= CHOICE {
       miles                       [0]          REAL,
       kilofeet                    [1]          REAL,
       kilometers                  [2]          REAL
       ...
}

SPECCode ::= PrintableString

SRInterofficeFacility ::= ENUMERATED {
   avoidance                                 (0),
   diversity                                 (1),
   avoidanceAndDiversity                     (2),
   selftHealingInterofficeFacility           (3),
   specialRoutingForInterofficeFacility      (4),
   routeOtherThanNormal                      (5),
   notApplicable                             (6)
       ...
}

SRLocation ::= ENUMERATED {
   cableOnly                                 (0),
   diversity                                 (1),
```

```
   disasterRecovery                        (2),
   routeOtherThanNormal                    (3),
   selfHealingLoop                         (4),
   alternateWireCenter                     (5),
   selfHealingLoopViaAlternateWireCenter   (6),
   selfHealingWireCenter                   (7),
   selfHealingAlternateWireCenter          (8),
   specialRoutingAtLocation                (9),
   notApplicable                           (10)
       ...
}

ServiceActivity ::= ENUMERATED {
       connectService              (0),
       disconnectService           (1),
       changeService               (2)
...
}

ServiceAlternatives ::= SET OF SEQUENCE {
       service                             [0]   ServiceDefinition
       OPTIONAL,
       quantity                            [1]   Quantity
              OPTIONAL,
       locations                           [2]   Locations
              OPTIONAL,
       availableDate                       [3]   Date
              OPTIONAL
       ...
}

ServiceAvailabilityError ::= SEQUENCE {
   serviceDefinitionNotFound               [0]   NULL          OPTIONAL,
   location1NotFound                       [1]   NULL          OPTIONAL,
   location2NotFound                       [2]   NULL          OPTIONAL,
   invalidAvailabilityDate                 [3]   NULL
       OPTIONAL,
   additionalText                          [4]   PrintableString     OPTIONAL
       ...
}

ServiceAvailabilityInformation ::= CHOICE {
   available                               [0]   NULL,
   notAvailable                            [1]   NULL,
   availableAlternatives                   [2]   ServiceAlternatives
       ...
}

ServiceAvailabilityReply ::= SEQUENCE {
   reply                                   [0]   ServiceAvailabilityInformation,
```

```
   additionalText                          [1]    PrintableString    OPTIONAL
       ...
}

ServiceAvailabilityRequest ::= SEQUENCE {
   service                                 [0]    ServiceDefinition,
   -- percentInterstateUsage is mandatory for access services
   percentInterstateUsage                  [1]    PercentInterstateUsage
       OPTIONAL,
   locations                               [2]    Locations
       OPTIONAL,
   quantity                                [3]    Quantity
       OPTIONAL,
   availabilityDate                        [4]    Date
       OPTIONAL
       ...
}

ServiceCodeDefinition ::= SEQUENCE {
       ncCode                              [0]    NCCode,
       nciCode                             [1]    NCICode,
       secNciCode                          [2]    NCICode            OPTIONAL,
       specCode                            [3]    SPECCode     OPTIONAL
       ...
}

ServiceDefinition ::= CHOICE {
-- ServiceCodeDefinition is now the only way of defining the service.
-- Additionally, an alternate method of defining service may be
-- presented later on. To accommodate both types of definitions the
-- following is presented.
   code       [0]    ServiceCodeDefinition,
   name       [1]    ServiceNameDefinition
       ...
}

ServiceLocationListR1 ::=  SET OF LocationInfo

ServiceNameDefinition  ::=  PrintableString


SpecialRouting ::= SEQUENCE {
-- specialRoutingAtLocation(9) value of primaryLocation indicates
-- special Routing at POP/PRILOC.
       primaryLocation                    SRLocation,
       interofficeFacility        SRInterofficeFacility,
       secondaryLocation          SRLocation
-- specialRoutingAtLocation(9) value of secondaryLocation indicates
-- special Routing at SECLOC.
       ...
```

```
}

TransferFeature ::= CHOICE {
       add           [0]    TransferFeatureOptions,
       disconnect    [1]    TransferFeatureOptions
       ...
}

TransferFeatureOptions ::= ENUMERATED {
       linesideOfPortCircuit             (0),
       regularPortOfCircuit              (1),
       standbyPortOfCircuit              (2),
       controlPathOfCircuit              (3),
       otherPort                         (4)
       ...
}

TransmissionLevel  ::= PrintableString

UnitNumber ::= PrintableString

UnnumberedAddress  ::= Address (
   WITH COMPONENTS {
              ... ,
              number        ABSENT
   }
)

END  -- EAO
```

REDACTED
FOR PUBLIC INSPECTION

REDACTED
FOR PUBLIC INSPECTION

REDACTED
FOR PUBLIC INSPECTION